

|  |
| --- |
| Data Cleaning and Merging Report *Brazilian-Ecommerce* |

# Data Cleaning and Merging Report for Olist Datasets

This report provides a detailed summary of the data cleaning process applied to the Olist datasets, which include **orders**, **order items**, **products**, **product category name translations**, **sellers**, **customers**, **order payments**, **order reviews**, and **geolocation** data. The objective was to ensure data quality by addressing inconsistencies, handling missing values, and standardizing formats across all files. Additionally, this report outlines the steps taken to merge these datasets into a unified dataset suitable for analysis, along with challenges encountered and recommendations for future improvements.

## Overview of Key Cleaning Steps for Each Dataset

Below is a breakdown of the key cleaning steps performed on each dataset, as derived from the Jupyter notebooks and cleaned output files located in the clean/Data\_Cleaned/ directory.

### 1. Customers Dataset (cleaned\_olist\_customers\_dataset.csv)

* **Removed Duplicates**: Eliminated duplicate entries based on customer\_id to ensure each customer is uniquely represented.
* **Handled Missing Values**: Filled missing customer\_city and customer\_state values with 'Unknown' to maintain data completeness.  
    
  **Data Type Standardization**: Converted customer\_zip\_code\_prefix to a string type to preserve leading zeros and ensure consistency with other datasets.

### 2. Geolocation Dataset (cleaned\_olist\_geolocation\_dataset.csv)

* **Aggregation**: Grouped data by geolocation\_zip\_code\_prefix, calculating the mean of geolocation\_lat and geolocation\_lng for average coordinates and selecting the most frequent geolocation\_city and geolocation\_state per zip code.
* **Duplicate Removal**: Removed duplicate entries to create a concise, representative dataset.
* **Missing Value Handling**: Addressed any missing values during aggregation to ensure reliable geographic data.

### 3. Order Items Dataset (cleaned\_olist\_order\_items\_dataset.csv)

* **Dropped Incomplete Rows**: Removed rows missing critical IDs (order\_id, product\_id, seller\_id) to maintain data integrity.
* **Validated Numerical Data**: Ensured price and freight\_value were non-negative by clipping values below zero.
* **Feature Engineering**: Added a total\_cost column by summing price and freight\_value for each order item.

### Order Payments Dataset (cleaned\_olist\_order\_payments\_dataset.csv)

* **Missing Value Handling**: Filled missing payment\_type entries with 'Unknown'.
* **Validated Numerical Data**: Ensured payment\_value was non-negative by clipping negative values.
* **Categorized Payments**: Simplified payment\_type into two categories: 'Card' (e.g., credit/debit cards) and 'Other' (e.g., vouchers, boleto) for easier analysis.

### 5. Order Reviews Dataset (cleaned\_olist\_order\_reviews\_dataset.csv)

* **Filled Missing Comments**: Replaced missing review\_comment\_title and review\_comment\_message with 'No Comment'.
* **Score Validation**: Confirmed review\_score values were within the valid range of 1 to 5.
* **Feature Engineering**: Calculated response\_time\_days as the difference between review\_answer\_timestamp and review\_creation\_date to measure responsiveness.

### 6. Orders Dataset (cleaned\_olist\_orders\_dataset.csv)

* **Dropped Incomplete Rows**: Removed rows with missing order\_id to ensure each order is uniquely identifiable.
* **Handled Missing Timestamps**: Filled missing values in timestamp fields (e.g., order\_purchase\_timestamp, order\_delivered\_customer\_date) with a placeholder future date ('2099-12-31') to maintain consistency.
* **Feature Engineering**: Calculated delivery\_time\_days as the difference between order\_delivered\_customer\_date and order\_purchase\_timestamp.

### 7. Product Category Name Translation Dataset (cleaned\_product\_category\_name\_translation.csv)

* **Dropped Incomplete Rows**: Removed rows missing product\_category\_name or product\_category\_name\_english to ensure complete translation mappings.
* **Text Standardization**: Converted English category names to lowercase and replaced spaces with underscores (e.g., "Home Appliances" → home\_appliances) for consistency.

### Products Dataset (cleaned\_products\_dataset.csv)

* **Dropped Incomplete Rows**: Removed rows with missing product\_id to ensure each product is uniquely identifiable.
* **Filled Missing Numerical Values**: Replaced missing values in numerical columns (e.g., product\_weight\_g, product\_length\_cm) with their respective column medians.
* ****Feature Engineering**:**
  + Calculated product\_volume\_cm3 by multiplying product\_length\_cm, product\_height\_cm, and product\_width\_cm.
  + Flagged products as is\_heavy if their product\_weight\_g exceeded the 75th percentile.

### 9. Sellers Dataset (cleaned\_sellers\_dataset.csv)

* **Dropped Incomplete Rows**: Removed rows with missing seller\_id to ensure each seller is uniquely identifiable.
* **Handled Missing Values**: Filled missing seller\_city and seller\_state with 'Unknown'.
* **Geographic Validation**: Verified seller\_state against Brazilian state codes and derived seller\_region for regional analysis.

## Merging Process

The merging process combined the cleaned datasets into a single, unified file (merged\_olist\_dataset.csv) to facilitate comprehensive analysis. The steps were as follows:

1. **Base Dataset**: Used the cleaned **orders dataset** as the foundation, as it contains core order information linking other datasets.
2. **Sequential Merges**:
   * **Order Items**: Merged with order\_id to include product and seller details per order.
   * **Products**: Merged with product\_id to add product-specific attributes.
   * **Category Translations**: Merged with product\_category\_name to incorporate English category names.
   * **Sellers**: Merged with seller\_id to include seller information.
   * **Customers**: Merged with customer\_id to add customer details.
   * **Order Payments**: Merged with order\_id to include payment information.
   * **Order Reviews**: Merged with order\_id to incorporate customer feedback.
3. **Geolocation Integration**:
   * Aggregated the **geolocation dataset** by geolocation\_zip\_code\_prefix to produce average coordinates and representative city/state values.
   * Merged this aggregated data with:
     + **Customers** using customer\_zip\_code\_prefix (renamed columns: e.g., customer\_lat, customer\_lng).
     + **Sellers** using seller\_zip\_code\_prefix (renamed columns: e.g., seller\_lat, seller\_lng).
4. ****Output**:** The final merged dataset, enriched with geographic coordinates, was saved as merged\_olist\_dataset.csv.

## Challenges and Recommendations

### Challenges

* **Data Type Consistency**: Ensuring uniform data types (e.g., converting zip\_code\_prefix to strings) was critical to avoid merging errors, particularly with fields prone to formatting issues like leading zeros.
* **Missing Value Handling**: Balancing data preservation with integrity required careful decisions, such as using placeholders ('Unknown', future dates) versus dropping rows.
* **Geographic Data Inconsistencies**: The geolocation dataset had multiple entries per zip code, necessitating aggregation to create a single, reliable entry per prefix.
* **Merge Complexity**: Linking multiple datasets with different keys increased the risk of misalignment or duplicate records.

## Conclusion

***The data cleaning and merging process for the Olist datasets has resulted in a high-quality, unified dataset (merged\_olist\_dataset.csv) ready for analysis. Each file was meticulously cleaned to address duplicates, missing values, and inconsistencies, while the merging process integrated all relevant information, including geographic coordinates for customers and sellers. Despite challenges such as data type alignment and geographic aggregation, the resulting dataset provides a robust foundation for exploring e-commerce trends, customer behavior, and geographic insights in the Brazilian market. Implementing the recommended validations and refinements will further enhance its reliability for future use.***